RESEARCH REPORT

SINGULAR VALUE DECOMPOSITION AND ITS APPLICATIONS

Shue-Sum Chow

No. 15 October 1980

Department of Mathematics

University of Canterbury
Christchurch
New Zealand

SINGULAR VALUE DECOMPOSITION

AND

ITS APPLICATIONS

by

SHUE-SUM CHOW

An essay submitted in support of an
application for a graduate scholarship
at the Australian National University

SINGULAR VALUE DECOMPOSITION AND ITS APPLICATIONS

SHUE-SUM CHOW

9 September, 1979.

1. Introduction

We shall consider a form of matrix factorization known as singular value decomposition (SVD) that is applicable to a general m-by-n matrix. Although still not widely known, the SVD is a powerful method in theoretical and computational analysis of problems involving matrices. It has already found applications in many areas and is of growing importance within and without the circle of numerical mathematics.

As the concept of the rank of a matrix, which is defined as the number of linearly independent columns of the matrix, is of fundamental importance in linear algebra, we often face the problem of determining the rank of a given matrix, or if the matrix is rank defective, that is, if the rank r of an m-by-n matrix is such that $r < \min(m,n)$. Theoretically, this can be found out easily by applying, for example, Gaussian Elimination (GE) to the given matrix and reducing it to a form from which the rank could easily be determined. In practice, if the size of the matrix is too big for us to solve by hand computation, we would invariably resort to a high speed automatic computer. However, as we will see later, because of the inherent limitation of the computer, instead of obtaining a solution to our problem, we must rephrase our problem so that we can interpret our result more sensibly.

The recent rapid advancement of digital computing technology, both in the areas of hardware and software development, has greatly enhanced the range of problems that can be solved in a reasonable amount of time.

However, the characteristic finite, fixed word length representation of data and the use of floating point arithmetic clearly imply inherent inaccuracy in the computations and also pose the problem of numerical instability, that is, the propagation of error in calculations. In almost all cases, the problem solved by the computer is a slightly perturbed version of the original problem. Naturally we would also expect the solution obtained to be a slightly perturbed version of the actual solution. Unfortunately this is often not the case. The presence of numerical instability, which arises from possible inaccuracies in input data, round-off error and truncation error both introduced in the calculation, may give rise to a totally unreliable result. When the error introduced propagates in increasing magnitude as subsequent operations are carried out, the departure of the computed solution from the actual solution may mean that we will not be able to infer the solution to the problem from the result we obtained. In fact, it is fair to say that a golden rule in numerical mathematics is: "Avoid all unstable methods, even though this may imply additional computational effort".

Returning to our discussion of finding the rank of a matrix by Gaussian Elimination with the aid of a digital computer. It is likely that input data for the elements of the matrix are not their exact values, and hence, even though the original matrix may be rank defective, the chance that the approximating matrix will also be is actually quite low. Also, because of the error introduced, the transformations involved in Gaussian Elimination may turn a matrix that is almost rank defective into one that is of full rank. Hence we ask, instead of whether the given matrix is defective in rank, if it is near a matrix of defective rank (in the sense of 2-norm or Frobenius norm).

Before proceeding any further, we point out that it is not always easy to recognize when a matrix is almost rank defective, even when the

matrix has a relatively simple structure as in the case of a triangular matrix which is resulting from Gaussian Elimination. For example, for a given m-by-n matrix A with $m \geq n$, GE with partial pivoting gives rise to a permutation matrix P, a unit lower triangular matrix L and an upper triangular matrix U such that

$$PA = L \begin{bmatrix} U \\ O \end{bmatrix}$$

If A is of rank $r \leq n$, then the last n-r rows of U will of course be zero rows. And so we would expect that when A is very close to a matrix of rank r, the entries in the last n-r rows would be small as compared with entries in the first r rows. This speculation is, however, incorrect. This can be easily seen from the following example:

If $A = (a_{ij})$ is an n-by-n matrix such that

$$a_{ij} = \begin{cases} 0 & \text{if } j < i \\ 1 & \text{if } j = i \\ -1 & \text{if } j > i \end{cases}$$

then clearly P = I, the identity matrix, L = I, and U = A. The matrix U has no small rows. However, it is not difficult to show that there exists a matrix E such that $\|E\|_2 \leq 2^{2-n}$ and U-E is singular, that is, U, and hence A, is close to a singular matrix. Indeed, if we let y be the vector such that

$$y_i = 2^{-i} \quad \text{for } i = 1(1)(n-1)$$
$$y_n = 2^{-n}$$

and $\underline{z}$ be the vector such that $\underline{z} = U\underline{y}$, the desired matrix E would be given by $\underline{z}\underline{y}^T/\underline{y}^T\underline{y}$. This can easily be verified for n = 4, in which case

$$\begin{bmatrix} 1 & -1 & -1 & -1 \\ 0 & 1 & -1 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$E = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{2}{11} & \frac{1}{11} & \frac{1}{22} & \frac{1}{22} \end{bmatrix}.$$

Although it is not always easy to know if a matrix is almost rank deficient, it is a simple matter to determine this condition for a diagonal matrix. While GE reduces a matrix to triangular form, the SVD reduces a matrix to diagonal form by premultiplications and postmultiplications of unitary matrices. Since only unitary matrices are used, the method is numerically stable and the final result will be near rank defective if and only if the original matrix is.

Although the existence of the SVD of general matrices was proved as early as 1939 by Eckart and Young [3], who incidentally also mentioned that Sylvester [11] had established the case for real square matrices, it does not receive very much attention until recently when Golub and Kahan [5] suggested its use in numerical analysis. Later on Golub adapted the QR algorithm for computing the eigenvalues and eigenvector of a symmetric matrix to the design of a numerically stable algorithm for computing the SVD of a matrix.

In the following sections we shall discuss the existence of a SVD, and describe the algorithm of Golub and Reinsch [6] and the problem of numerical stability. Some applications of the SVD will be presented to demonstrate its wide range of applicability.

II. Existence, construction and stability of SVD

We shall assume that the given matrix A is m - by - n, with $m \leqslant n$. If this is not the case we can either work with the Hermitian of A, A^H,

or adjoin n - m rows of zeros to the matrix A. Note that we do not actually require any storage or arithmetic operations on the elements of the adjoined rows since they can be handled implicitly in a computer program.

Let $\lambda_j(B)$, j = 1(1)p, denote the eigenvalues of a p - by - p matrix B such that $\lambda_i(B) \leqslant \lambda_j(B)$ if $i \geqslant j$. Since both A^HA and AA^H are positive semi-definite, we have $\lambda_j(A^HA) \geqslant 0$ for j = 1(1)n and $\lambda_j(AA^H) \geqslant 0$ for j = 1(1)m. It is also clear that A^HA and AA^H have the same set of non-zero eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_r\}$, where $r \leqslant n$.

DEFINITION. The singular values of A, denoted by σ_j, j = 1(1)n are defined as

$$\sigma_j = \begin{cases} +\sqrt{\lambda_j} & , \quad j = 1(1)r \\ 0 & , \quad j = (r+1)(1)n \, . \end{cases}$$

That is, the singular values of A are the positive square root of the eigenvalues of A^HA. We note that if B is Hermitian and has eigenvalues $\lambda_i(B)$, i = 1(1)p, then the singular values of B are $|\lambda_i(B)|$, i = 1(1)p. Since $A^HA = AA^H$, we can readily deduce that A and A^H have the same singular values.

Recall that the eigenvalues of a matrix are invariant under similarity transformations. We are interested to find a class of transformations on A which leave the singular values unaltered. It turns out that the desired transformation is a unitary equivalence transformation, that is, if P is an m - by - m unitary matrix, and Q an n - by - n unitary matrix, then PAQ has the same singular values as A. Since we can easily determine the rank of a matrix if it is diagonal, it is natural to ask if there exist unitary matrices U,V which, upon premultiplication and postmultiplication respectively on A, would yield a diagonal matrix whose elements on the diagonal are precisely the singular values of A. The answer to the question is affirmative and is presented in the following

Theorem (Existence of SVD):

For each m - by - n matrix A, there exist an m - by - m unitary matrix U, an n - by - n unitary matrix V, such that

$$U^H A V = \Sigma ,$$

where $\Sigma = \begin{bmatrix} D \\ \hline 0 \end{bmatrix}$ and D is a diagonal matrix $\mathrm{diag}(\sigma_1,\sigma_2,\ldots,\sigma_n)$, where σ_i's are the singular values of A.

Proof. Let $r \leqslant n$ be such that $\sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$.

From the definition of singular values, it is clear that $\sigma_1^2,\sigma_2^2,\cdots,\sigma_r^2$ are the non-zero eigenvalues of $A^H A$. Let $\underline{v}_1,\underline{v}_2,\cdots,\underline{v}_r$ be the corresponding orthonormal eigenvectors. Since $A^H A$ is Hermitian, it possesses a complete set of eigenvectors and we could find (n - r) orthonormal vectors in the null space of $A^H A$ such that they are orthogonal to each of the vector $\underline{v}_i$, i = 1(1)v. We shall call these (n - r) vectors $\underline{v}_{r+1},\underline{v}_{r+2},\cdots,\underline{v}_n$, since they clearly are eigenvectors corresponding to the zero eigenvalue of $A^H A$. Let $V = [\underline{v}_1,\underline{v}_2 \cdots \underline{v}_n]$. By the orthonormality of $\underline{v}_i$, i = 1(1)n, V is clearly unitary. Now we define

$$\underline{u}_i = A\underline{v}_i/\sigma_i \quad \text{for} \quad i = 1(1)r .$$

Now $AA^H\underline{u}_i = \sigma_i^2\underline{u}_i$, i = 1(1)r, and, also for i,j such that $1 \leqslant i,j \leqslant r$,

$$\underline{u}_i^H\underline{u}_j = \frac{\sigma_j}{\sigma_i} v_i^H\underline{v}_j = \delta_{ij}, \quad \text{the Kronecker delta, by the}$$

orthonormality of $\underline{v}_i$'s. Thus we deduce that the set of orthonormal vectors $\underline{u}_i$, i = 1(1)r, are the eigenvectors of AA^H corresponding to the eigenvalues σ_i^2, i = 1(1)r, (of $A^H A$). Using an argument similar to that above, we construct a unitary matrix $U = [\underline{u}_1,\underline{u}_2\cdots\underline{u}_m]$, where $\underline{u}_i$, i = r+1(1)m are linearly independent orthonormal null vectors of AA^H. Now $A^H A\underline{v}_i = 0$ implies that $\underline{v}_i^H A^H A\underline{v}_i = 0$ which in turn implies $A\underline{v}_i = 0$, for i = r+1(1)m. Also

$$\sigma_i^2 \underline{u}_i = AA^H \underline{u}_i = \sigma_i A\underline{v}_i \quad , \text{ for } i = 1(1)r,$$

which gives $A\underline{v}_i = \sigma_i \underline{u}_i$ for $i = 1(1)r$.

Thus we obtain $U^H AV = U^H [A\underline{v}_1 \; Av_2 \; \ldots \; A\underline{v}_n]$

$$= U^H [\sigma_1 \underline{u}_1 \; \sigma_2 \underline{u}_2 \ldots \sigma_r \underline{u}_r \; 0 \ldots 0]$$

$$= \begin{bmatrix} D \\ \hline 0 \end{bmatrix} = \Sigma \; . \qquad \blacksquare$$

From the above proof, it is easy to see that there is a certain degree of freedom in the choice of U and V. If A^HA has a multiple eigenvalue $\sigma^2 > 0$, the corresponding columns of V may be chosen as any orthonormal basis for the space spanned by the eigenvectors corresponding to σ^2. Once $\underline{v}_1, \ldots, \underline{v}_r$ are chosen, the corresponding vectors $\underline{u}_1, \underline{u}_2, \ldots, \underline{u}_r$ will be fixed. However, we would still have some form of arbitrariness in the choice of the remaining column vectors.

The constructive nature of the proof immediately suggests an algorithm for factorizing A. However, the possible multiplicity of the eigenvalues of A^HA could complicate our determination of the eigenvalues and eigenvectors. Also it is computationally inefficient to perform the algorithm, both in terms of the amount of computational effort and of accuracy, since if any σ_i are such that

$$\sigma_i \leqslant 2^{-t/2} \; \sigma_1 \quad ,$$

where t is the number of digits used in floating point arithmetic computation with rounding, these small singular values will be reduced to round-off error proportion in the squaring process. [7]

We shall call the column vectors of V the right singular vectors of A and the column vectors of U the left singular vectors of A. It is not difficult to see that the 2-norm of A, $\|A\|_2$, or the spectral norm of A, is just σ_1, and that the Frobenius norm of A, $\|A\|_F$, is given by

$$\|A\|_F^2 = \sigma_1^2 + \sigma_2^2 + \ldots + \sigma_r^2 .$$

Before proceeding to describe the algorithm of Golub, we would like to justify what was said before, that the singular values of a matrix are invariant under unitary equivalence transformation. Indeed if we decompose A into its SVD form $A = U\Sigma V^H$, then

$$P^H A Q = (P^H U)\ \Sigma\ (Q^H V)^H \ .$$

Now if P and Q are unitary, then, since the product of two unitary matrices is unitary, therefore $P^H U$ and $Q^H V$ are both unitary and hence $P^H AQ$ has the same singular values as A. This invariance property is also of great importance in the understanding of the algorithm of Golub, which we shall now describe.

The algorithm of Golub is divided into two stages. Stage one is a direct scheme to reduce A to upper bidiagonal form by a sequence of unitary elementary transformations. Stage two is an iterative scheme to reduce the bidiagonal matrix to diagonal form by successive pre-multiplications and post-multiplications with unitary matrices. The second stage is closely related to the QR algorithm and hence possesses certain pleasing properties of the QR algorithm.

In the first stage we attempt to reduce A by a sequence of at most $2n - 2$ Householder transformation matrices to a bidiagonal form $\begin{bmatrix} J \\ 0 \end{bmatrix}$, where $J = (J_{ij})$ is such that

$$J_{ij} = 0 \quad \text{for } i > j \quad \text{or } j-1 > i, \text{ with } 1 \leqslant i,\ j \leqslant n.$$

For a given vector $\underline{Z}$ with m elements, if we want to zero the elements in rows $k + 1$ through m, we construct a Householder transformation $P = I - 2\underline{x}\,\underline{x}^H$, such that

$$\underline{x} = \begin{pmatrix} \underline{0} \\ Z_k \pm \alpha \\ Z_{k+1} \\ \cdots \\ Z_m \end{pmatrix} \Big/ \gamma$$

where $\alpha = \sqrt{(Z_k^2 + Z_{k+1}^2 + \ldots + Z_m^2)}$ and $\gamma = \sqrt{2(\alpha^2 \pm \alpha Z_k)}$, the sign taken to ensure addition in γ.

Now if we let $A^{(1)} = A$, and construct two finite sequences of Householder transformations $P^{(k)}$, $k = 1(1)n$, and $Q^{(k)}$, $k = 1(1)(n-2)$ such that

$$\left.\begin{aligned} A^{(2k)} &= P^{(k)} A^{(2k-1)} , \\ A^{(2k+1)} &= A^{(2k)} Q^{(k)} , \end{aligned}\right\} \quad \text{for } k = 1(1)\ (n-2)$$

$$\begin{aligned} A^{(2n-2)} &= P^{(n-1)} A^{(2n-3)} , \\ A^{(2n-1)} &= P^{(n)} A^{(2n-2)} , \end{aligned}$$

where $P^{(k)}$ would annihilate the elements in rows k+1 through m of column k of the matrix it premultiplies and $Q^{(k)}$ would annihilate the elements in columns k+2 through n of row k of the matrix it postmultiplies, then

$$P^{(n)}(P^{(n-1)}(\ldots\ ((P^{(1)}A)Q^{(1)})\ldots\ Q^{(n-2)})) = \begin{bmatrix} J \\ 0 \end{bmatrix} .$$

Note that for $m = n$, $P^{(m)} = I$, the identity matrix.

We illustrate below the effect of premultiplications and postmultiplications when $m = 5$, $n = 3$. Changed elements are underlined.

$$\begin{bmatrix} x & x & x \\ x & x & x \\ x & x & x \\ x & x & x \\ x & x & x \end{bmatrix} \xrightarrow{P^{(1)}} \begin{bmatrix} \underline{x} & \underline{x} & \underline{x} \\ \underline{0} & \underline{x} & \underline{x} \\ \underline{0} & \underline{x} & \underline{x} \\ \underline{0} & \underline{x} & \underline{x} \\ \underline{0} & \underline{x} & \underline{x} \end{bmatrix} \xrightarrow{Q^{(1)}} \begin{bmatrix} x & \underline{x} & \underline{0} \\ 0 & \underline{x} & \underline{x} \\ 0 & \underline{x} & \underline{x} \\ 0 & \underline{x} & \underline{x} \\ 0 & \underline{x} & \underline{x} \end{bmatrix} \xrightarrow{P^{(2)}} \begin{bmatrix} x & x & 0 \\ 0 & \underline{x} & \underline{x} \\ 0 & \underline{0} & \underline{x} \\ 0 & \underline{0} & \underline{x} \\ 0 & \underline{0} & \underline{x} \end{bmatrix}$$

$$\xrightarrow{P^{(3)}} \begin{bmatrix} x & x & 0 \\ 0 & x & x \\ 0 & 0 & \underline{x} \\ 0 & 0 & \underline{0} \\ 0 & 0 & \underline{0} \end{bmatrix} = \begin{bmatrix} J \\ 0 \end{bmatrix}$$

Since Householder transformations are unitary, the invariance property of singular values implies that J and A have the same set of singular values. In practice, very often only the singular values of A are required and in such cases there is no need to store the transformations.

$\underline{x}^{(k)}$, $\underline{y}^{(k)}$ are such that

$$P^{(k)} = I - 2\underline{x}^{(k)}\underline{x}^{(k)H} \quad , \quad \underline{x}^{(k)H}\underline{x}^{(k)} = 1 \ ;$$
$$Q^{(k)} = I - 2\underline{y}^{(k)}\underline{y}^{(k)H} \quad , \quad \underline{y}^{(k)H}\underline{y}^{(k)} = 1 \ .$$

We also note that if $m \gg n$, we can improve the efficiency by first reducing A to upper triangular form by pre-multiplications and then reducing this form to bidiagonal form. By doing this we save all the arithmetic operations on the elements below the diagonal while we are annihilating elements row by row to transform the matrix to bidiagonal form.

The second stage of the algorithm is to construct a sequence of matrices $(J^{(i)})$, with $J^{(1)} = J$, such that $J^{(i)}$ is upper bidiagonal and $J^{(i)} \to \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ as $i \to \infty$. Since J has the same singular values as A, we would like to find unitary matrices P,Q such that

$P^H JQ = D$, a diagonal matrix whose diagonal entries are the singular values of A. The construction of P,Q is as follows:

Let $J^{(1)} = J$,

for $k = 1,2,3,\ldots$,

construct P_k, Q_k in the manner to be described below, so that $J^{(k+1)} = P_k^H J^{(k)} Q_k$ is bidiagonal, and is closer to a diagonal matrix then $J^{(k)}$, in the sense that the magnitude of the superdiagonal elements of $J^{(k+1)}$ will be less than that of $J^{(k)}$, then

$$P = P_1 P_2 P_3 \ldots .$$

and $$Q = Q_1 Q_2 Q_3 \ldots . \quad .$$

Let $\bar{J}^H \bar{J}$ be denoted by $\bar{M}$ and $J^H J$ by M, where we have dropped the superscripts of $J^{(k)}$ and $J^{(k+1)}$ and simply write J and $\bar{J}$ respectively for a typical iteration step k. We observe that both M and $\bar{M}$ are Hermitian and tridiagonal and that $\bar{M}$ is unitary similar to M, that is $\bar{M} = Q^H M Q$. Hence M and $\bar{M}$ have the same set of singular values which is also the set of eigenvalues of M and $\bar{M}$. This strongly suggests to us that we apply the QR

algorithm to find the eigenvalues of M. However, as we have remarked before, this is not desirable since certain small singular values may be reduced to round-off error proportions by the squaring effect.

Knowing that we should not work directly with M, we would like to know if it is possible to work with J in such a manner that the result produced is directly related to that obtained by using the QR algorithm with shifts on M, a tridiagonal and hence upper Hessenberg matrix. The answer is in the affirmative. Indeed, recall that a typical step of the QR algorithm with shift parameter μ, when applied to M, would produce a tridiagonal $\hat{M}$ via

$$M - \mu I = \hat{Q}\hat{R} \quad ,$$

$$\hat{M} = \hat{R}\hat{Q} + I \quad ,$$

where $\hat{Q}$ is unitary and $\hat{R}$ upper triangular. Noting that $\hat{M} = \hat{Q}^H M \hat{Q}$, it is clear that if we choose Q to be $\hat{Q}$, and find P to reduce MQ to bidiagonal form $\bar{J}$, then $\bar{M} = \hat{M}$.

Before finding ways of constructing Q so that it is the same $\hat{Q}$ we would have obtained if the QR algorithm is used, we first note that if

$$J = \begin{bmatrix} q_1 & \ell_2 & & & \\ & q_2 & \ell_3 & & \\ & & \cdots & & \\ & & & q_{n+1} & \ell_n \\ & & & & q_n \end{bmatrix} ,$$

and if any ℓ_i is zero, we can partition J into smaller blocks that can be dealt with independently. If we have $q_k = 0$, with $q_j \neq 0$ and $\ell_j \neq 0$, $j = (k+1)(1)n$, premultiply J by $(n-k)$ Given's (plane) rotations $R_{k(k+1)}, R_{k,(k+2)}, \ldots, R_{kn}$, to get a matrix which we can partition it into smaller blocks that can be treated separately:

$$R_{kn}\cdots R_{k(k+1)}J = \left[\begin{array}{ccc|ccc} q_1 & \ell_2 & & & & \\ & \ddots & & & & \\ & q_{k-1} & \ell_k & & & \\ & & 0 & 0 & & \\ \hline & & & q'_{k+1} & \ell'_{k+2} & \\ & & & & \ddots & \\ & & & & q'_{n-1} & \ell'_n \\ & & & & & q'_n \end{array}\right] = \left[\begin{array}{c|c} J_1 & 0 \\ \hline 0 & J_2 \end{array}\right]$$

with $\ell'_j \neq 0$, $j = (k+2)(1)n$ and $q'_j \neq 0$, $j = (k\_1)(1)n$. The rotation R_{kj} annihilates the entry at the intersection of row k and column j, $j = (k+1)(1)n$. We illustrate below the effect of premultiplication for $k = 3$ and $n = 6$.

$$\begin{bmatrix} x & x & & & & \\ 0 & x & x & & & \\ & 0 & x & & & \\ & & x & x & & \\ & & & x & x & \\ & & & & & x \end{bmatrix} \xrightarrow{R_{34}} \begin{bmatrix} x & x & & & & \\ & x & x & & & \\ & 0 & \underline{0} & + & & \\ & & \underline{x} & \underline{x} & & \\ & & & x & x & \\ & & & & & x \end{bmatrix} \xrightarrow{R_{35}} \begin{bmatrix} x & x & & & & \\ & x & x & & & \\ & & 0 & 0 & \underline{0} & + \\ & & & x & x & \\ & & & \underline{x} & \underline{x} & \\ & & & & & x \end{bmatrix} \xrightarrow{R_{36}} \begin{bmatrix} x & x & & & & \\ & x & x & & & \\ & & 0 & 0 & 0 & \\ & & & x & x & \\ & & & & x & \\ & & & & & x \end{bmatrix}$$

Since the lower right corner element of J_1 is zero, it has at least one zero singular value. This fact can be used to eliminate ℓk, by performing a sequence of rotations

$$J'_1 = J_1 R_{(k-1)k} \cdots R_{1k} ,$$

where R_{ik} is constructed to zero the entry at position (i,k), when the algorithm is applied to operate on J_1.

In the light of the partitioning processes described above, it is clear that we can assume J has non-zero diagonal and superdiagonal elements. The construction of the matrix Q is now possible with the aid of the following.

Theorem: Let A, B, U be p-by-p matrices such that U is unitary, $B = (b_{ij})$ is upper Hessenberg with $b_{i+1,i} \neq 0$ for $i = 1(1)(n-1)$. If $A = UBU^H$, then both B and U are uniquely determined, up to a constant factor of absolute value unity, by the first column of U, or, equivalently, by the first row of U^H.

The proof of the theorem can be found in [10]. According to the theorem, if the first column of $\hat{Q}$ is known, then we can construct $\hat{Q}$ and $\hat{M}$. This is an important piece of information because in practice, when the QR algorithm is applied to M, $\hat{Q}$ is usually expressed as the Hermitian of the product of (n-1) Given's rotations, that is

$$\hat{Q}^H = R_{(n-1)n} \cdots R_{23} R_{12} ,$$

where R_{pq} is a plane rotation of Given's type in the plane (p,q), with the four strategic elements denoted by

$$\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}.$$

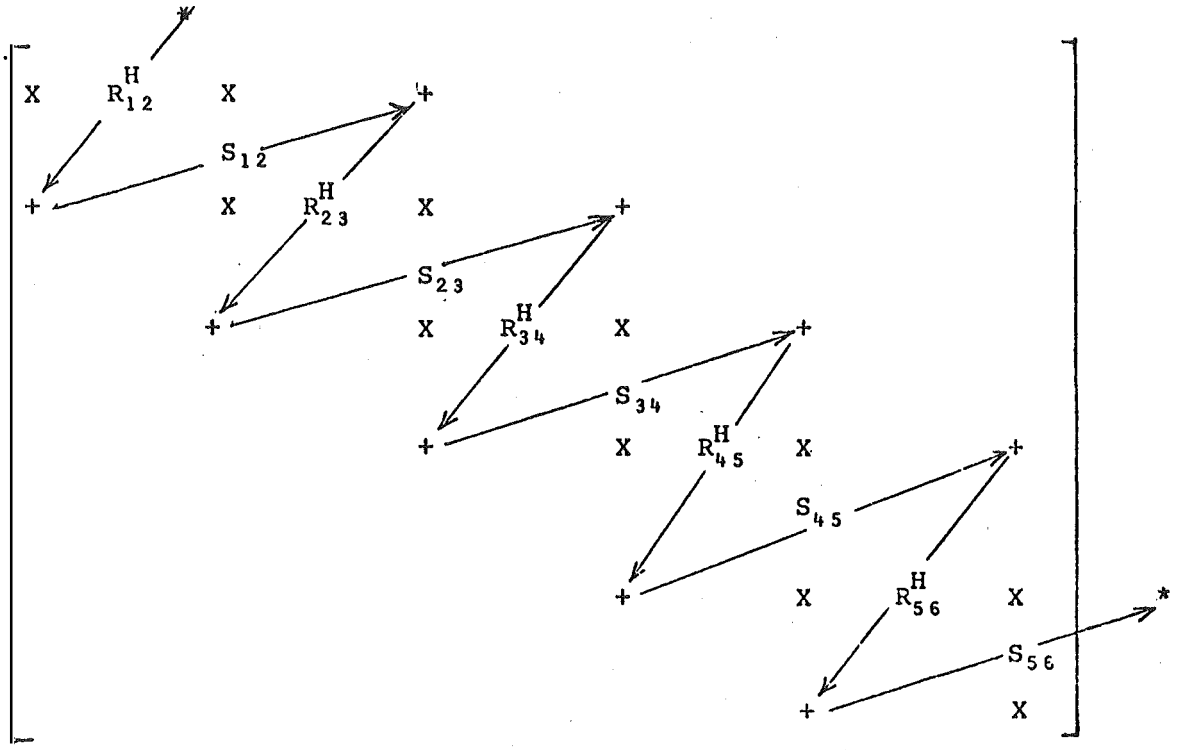
It is not difficult to see that the first row of $\hat{Q}^H$ is the same as the first row of R_{12}. Since R_{12} is completely determined if we know the first column of M, which can easily be deduced from J, we are now in a position to construct P and Q so that $\bar{J} = P^H J Q$ is bidiagonal and $\bar{M} = \hat{M}$.

After obtaining R_{12}, we found that on postmultiplying J by R_{12}^H, a new non-zero element is generated and that another (2n-3) plane rotations are required to restore the matrix to bidiagonal form $\bar{J}$, that is

$$\bar{J} = S_{(n-1)n}(\cdots S_{23}((S_{12}(JR_{12}^H))R_{23}^H)\cdots R_{(n-1)n}^H).$$

$S_{p(p+1)}$ and $R_{p(p+1)}$ are plane rotations such that $S_{p(p+1)}$ annihilates the entry of position (p+1,p), and generates an entry of position (p,p+2), for $p = 1(1)(n-2)$; $R_{p(p+1)}^H$ annihilates the entry of position (p-1,p+1) and generates an entry of position (p+1,p), for $p = 2(1)(n-1)$. R_{12}^H does not annihilate any entry but generates a new entry at position (2,1) and $S_{(n-1)n}$

annihilates the entry at (n,n-1) but generates nothing. This 'chasing' process is illustrated below for the case of n = 6.



Here "+" indicates the new non-zero element generated by the plane rotation. Since the first row of R_{12} is the same as the first row of $Q^H = R_{(n-1)n}\ldots R_{23}R_{12}$, then $Q = \hat{Q}$. Also $P^H = S_{(n-1)n}\ldots S_{23}S_{12}$ is clearly unitary. Hence we have found a way of computing $\bar{J}$ such that $\bar{M} = \hat{M}$, the matrix that would be generated by the QR algorithm if we had used it on M. We now generate the sequence by repeating the process, resetting $\bar{J}$ to J, and partitioning the matrix into smaller blocks if an intermediate q_r or ℓ_r becomes negligible. If ℓ_n is negligible, we accept q_n as a singular value and work with a matrix of order n - 1. The successive M matrices of the QR algorithm with shifts, when applied to symmetric tridiagonal matrices, converge globally with at least quadratic convergence to a diagonal matrix [12] . Then since any Hermitian matrix can be transformed by Given's plane rotations to a symmetric tridiagonal matrix, and also all the eigenvalues of M are real, the successive J matrices tend to diagonal form.

The algorithm described above is very stable. In fact, Wilkinson has shown that "the computed [matrix] $\bar{\Sigma}$ is exactly related to some neighbouring

A+E [matrix] [E being a perturbation matrix]. In practice when the statistical distribution is taken into account it is highly improbable that $\|E\|_2$ will attain, say, $n2^{-t}\|A\|_2 = n2^{-t}\sigma_1$ [where t is the number of digits used in floating point binary computation with rounding.] Hence the computed σ_i are extremely reliable." [7]

Another remarkable feature is that the singular values themselves are also very stable with respect to changes in the matrix elements. This can be seen from the following theorem regarding the perturbation of singular values.

Theorem: Let A,B, and E be m-by-n matrices with B = A + E. Let k - be min(m,n), then

$$\text{(i)} \quad |\sigma_i(B) - \sigma_i(A)| \leqslant \sigma_1(E) = \|E\|_2 \quad , \; i = 1(1)k.$$

$$\text{(ii)} \quad \sum_{i=1}^{k} [\sigma_i(B) - \sigma_i(A)]^2 \leqslant \sum_{i=1}^{k} \sigma_i(E)^2 = \|E\|_F^2 \; .$$

The proof of the above theorem can be found in [8].

III. Applications

The SVD, equipped with its rigorous theoretical background, has found applications in many areas. We shall discuss briefly its application in solving least square problems, estimating condition numbers and in digital images processing.

We are now in a position to determine the numerical rank of a given m-by-n matrix A with $m \geqslant n$. Assume that the accuracy of the elements of A is of order . Let the SVD of A be $U\Sigma V^H$, where $\Sigma = \begin{bmatrix} D \\ 0 \end{bmatrix}$ and $D = \text{diag}(\sigma_1,\sigma_2,\ldots,\sigma_n)$. σ_i are the singular values of A. We shall say A is close to a matrix of rank r if

$$\sigma_{r+1}^2 + \sigma_{r+2}^2 + \ldots + \sigma_n^2 < \varepsilon^2 \; .$$

Let $D' = \text{diag}(\sigma_1,\sigma_2,\ldots,\sigma_r,0,\ldots,0)$ and let

$$\Sigma' = \begin{bmatrix} D' \\ 0 \end{bmatrix} \quad , \quad A' = U\Sigma' V^H \; ,$$

then A' is clearly of rank r, using the usual definition.

Also, $\|A - A'\|_F = (\sigma_{r+1}^2 + \sigma_{r+2}^2 + \ldots + \sigma_n^2)^{\frac{1}{2}} < \varepsilon$. This justifies our statement that A is close to a matrix of rank r. In fact, A' is a matrix of rank k nearest to A in the Frobenius norm, that is

$$\|A - A'\|_F = \min_{\text{rank}(B)=r} \|A - B\|_F .$$

The proof of the above statement can be found in [10].

Least squares data fitting

Frequently we are given in data points

$$(t_i, y_i) \quad i = 1(1)m ,$$

and we are required to establish a linear mathematical model for these data points. That is, we need to find coefficients $c_1, c_2, \ldots, c_n$ so that, for some given basis functions $\phi_1, \phi_2, \ldots, \phi_n$, the model

$$y(t) = c_1\phi_1(t) + c_2\phi_2(t) + \ldots + c_n\phi_n(t)$$

fits the data in the following sense.

Let the residual at the ith data point be

$$r_i = \sum_{j=1}^{n} c_j\phi_j(t_i) - y_i .$$

We shall say the model fits the data if we can choose c_j so that the least squares criterion is satisfied. By this, we require c_j be chosen to minimize the sum of the squares of the residuals,

$$\underset{c_j, 1 \leq j \leq n}{\text{minimize}} \sum_{i=1}^{m} r_i^2 .$$

Note that the linear mathematical model may be nonlinear in nature, since the basis functions may be nonlinear in t. Also, it may not be the actual functional relationship that relates t_i and y_i. This is especially the case when m > n since we would then have an overdetermined problem of choosing c_j, and it is unlikely that the model would interpolate the data. However, if the model does represent the actual functional relationship, the minimum we

obtained from the least squares method will be zero, which indicates the model fits the data exactly.

We remark that the least squares criterion does not imply that the set of coefficients $\{c_j\}$ could be chosen uniquely. It can be verified that the sum of the squares of the residuals is unchanged if any multiple of the γ_j for which

$$\sum_{j=1}^{n} \gamma_j \phi_j(t_i) = 0 \quad , \quad i = 1(1)m \quad ,$$

is added to c_j , $j = 1(1)n$, provided that the basis functions are linearly dependent at the data points.

Let A be an m-by-n matrix whose elements are

$$a_{ij} = \phi_j(t_i).$$

Let $\underline{b}$ be an m-vector with elements y_i and $\underline{x}$ be an n-vector with elements c_j. We now rephrase our problem as:

'Given a real m-by-n matrix A of rank $r \leqslant \min(m,n)=n$, and a real m-vector $\underline{b}$, find a real n-vector $\underline{x}$ such that the 2-norm of $A\underline{x} - \underline{b}$ is minimized, that is

$$\underset{\underline{x}}{\text{minimize}} \; \|A\underline{x} - \underline{b}\|_2 '.$$

Because of the non-uniqueness discussed above, we impose the additional condition that $\|\underline{x}\|_2$ be minimized as well. Now

$$\|A\underline{x} - \underline{b}\|_2 = \|U\Sigma V^T\underline{x} - \underline{b}\|_2 = \|U(\Sigma V^T\underline{x} - \underline{d})\|_2 = \|\Sigma\underline{z} - \underline{d}\|_2,$$

where $U\Sigma V^T$ is the SVD of A, $\underline{z} = V^T\underline{x}$ and $\underline{d} = U^T\underline{b}$.

Clearly if we can find $\underline{z}$ which minimizes $\|\Sigma\underline{z} - \underline{d}\|_2$, then $\underline{x} = V\underline{z}$ would minimize $\|A\underline{x} - \underline{b}\|_2$. Moreover, since $\|\underline{x}\|_2 = \|\underline{z}\|_2$, $\|\underline{x}\|_2$ is minimized if $\|\underline{z}\|_2$ is. Since A is of rank r, $\Sigma = \begin{bmatrix} D \\ 0 \end{bmatrix}$ where $D = \text{diag}(\sigma_1,\sigma_2,\ldots,\sigma_r,0,\ldots,0)$.

So $\|\Sigma\underline{z} - \underline{d}\|_2^2 = (\sigma_1 z_1 - d_1)^2 + \ldots + (\sigma_r z_r - dr)^2 + d_{r+1}^2 + \ldots + d_n$.

Hence $\|\Sigma\underline{z} - \underline{d}\|_2^2$ achieves its minimum when $z_i = {}^{d_i}/_{\sigma_i}$, $i = 1(1)r$, irrespective of the values of $z_{r+1},\ldots, z_n$. Therefore the vector $\underline{z}$ for which $\|z\|_2$ is minimized subject to the condition that $\|\Sigma\underline{z} - \underline{d}\|_2$ is a minimum is given by

$$z_i = \begin{cases} d_i/\sigma_i & \text{for} \quad i = 1(1)r \\ 0 & \text{for} \quad i = (r+1)(1)n \ . \end{cases}$$

The solution $\underline{x}$ to the original problem is given by $\underline{x} = V\underline{z}$.

In practice, we would first establish the SVD of A, then determine the rank r of A as described above, and work with A', the matrix of rank r that is closest to A in the Frobenius norm.

The notion of pseudo inverse is also of importance in the solution of least squares problems. Recall that A^+ is a pseudo inverse of A if

(i) $AA^+A = A$

(ii) $A^+AA^+ = A^+$

(iii) $AA^+ = (AA^+)^H$

(iv) $A^+A = (A^+A)^H$.

It is easy to verify that if $D = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r, 0, \ldots, 0)$, then $D^+ = \text{diag}(\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_r^{-1}, 0, \ldots, 0)$ is a pseudo inverse of D. Hence the pseudo inverses of $\Sigma = \begin{bmatrix} D \\ 0 \end{bmatrix}$ and $A = U\Sigma V^H$ are $\Sigma^+ = [D^+ | 0]$ and $A^+ = V^H \Sigma^+ U$ respectively. Hence if we have already decomposed A into its SVD form, it is relatively easy to find its pseudo inverse A^+.

Condition Number Estimation

In solving the least squares problems, if the matrix A is found to be of rank r, and σ_r is small but not too small to be regarded as negligible, then σ_r^{-1} will be correspondingly large and large round-off errors may be induced in subsequent calculations involving σ_r^{-1} . Thus some measure of accuracy is required to indicate the reliability of the computed result.

For an invertible matrix A, the condition number K(A) provides such a measure. The condition number is defined as

$$K(A) = \|A\| \ \|A^{-1}\|$$

where $\|\cdot\|$ is a norm. If A is not invertible, we may consider $K(A) = \infty$. The condition number is closely related to sensitivity of the solution of

the system of linear equations

$$A\underline{x} = \underline{b}$$

with respect to perturbation E in A or to perturbation $\underline{e}$ in $\underline{b}$. It can be shown that if

$$(A+E)\underline{y} = \underline{b}$$

then

$$\|\underline{y} - \underline{x}\| / \|\underline{y}\| \leqslant \varepsilon\, K(A) \ ,$$

where $\varepsilon = \|E\| / \|A\|$ is the relative error in A, and that if

$$A\underline{y} = \underline{b} + \underline{e} \quad , \quad \text{with} \quad \|\underline{e}\| < \varepsilon \|\underline{b}\| \ ,$$

then

$$\|\underline{y} - \underline{x}\| / \|\underline{x}\| \leqslant \varepsilon\, K(A) \ .$$

Although the error bounds for the perturbation in A and for the perturbation in $\underline{b}$ are essentially the same, the probability that the bound is attained is quite different for the two different types of perturbation. It turns out that the chance of attaining the bound is higher in the case of perturbation in A than that in the case of perturbation in $\underline{b}$. This can be seen easier if we look at the SVD of A.

Let $U\Sigma V^H$ be the SVD of A. Then

$$A\underline{v}_i = \sigma_i \underline{u}_i \quad , \quad A^T \underline{u}_i = \sigma_i \underline{v}_i \ .$$

Since U is orthogonal, we expand $\underline{b}$ and $\underline{e}$ as

$$\underline{b} = \|\underline{b}\| \sum_{i=1}^{n} \alpha_i \underline{u}_i \ ,$$

$$\underline{e} = \varepsilon \|\underline{b}\| \sum_{i=1}^{n} \beta_i \underline{u}_i$$

with $\Sigma \alpha_i^2 = 1$, $\Sigma \beta_i^2 = 1$ for consistency. It is not difficult to see that

$$\underline{x} = \|\underline{b}\| \sum_{i=1}^{n} (\alpha_i / \sigma_i) \underline{v}_i \ ,$$

$$\underline{y} - \underline{x} = \varepsilon \|\underline{b}\| \sum_{i=1}^{n} (\beta_i / \sigma_i) \underline{v}_i \ .$$

Therefore, in order to attain the bound $\|\underline{y} - \underline{x}\| / \|\underline{x}\| = \varepsilon\, K(A)$, we need to have

$$\underline{b} = \|\underline{b}\| \underline{u}_1 \quad , \quad \underline{e} = \varepsilon \|\underline{b}\| \underline{u}_n \ .$$

Clearly, the probability of having the above situation is low, hence the probability that the bound will be attained is low.

Also, since

$$\|\underline{x}\|/\|\underline{b}\| = \left[\sum_{i=1}^{n} (\alpha_i/\sigma_i)^2\right]^{\frac{1}{2}},$$

this ratio will be of order σ_n^{-1} unless α_n, which gives the component of $\underline{b}$ in the direction of $\underline{u}_n$, is exceptionally small. Hence if K(A) is large, the probability that $\|\underline{x}\|/\|\underline{b}\|$ will give a good estimate of σ_n^{-1} is quite high. This immediately suggests the following two step process for estimating σ_n^{-1}.

Randomly choose a vector $\underline{b}$, solve the systems of equations

$$A^T\underline{x} = \underline{b} \quad , \quad A\underline{y} = \underline{x}$$

and take $\|\underline{y}\|/\|\underline{x}\|$ as an estimate of σ_n^{-1} .

The reason for estimating σ_n^{-1} is that we would like to have an estimate of K(A) which will give at least a reliable indication of its order of magnitude, and since $\|A\|_2 = \sigma_1$, $\|A^{-1}\|_2 = \sigma_n^{-1}$, therefore $K_2(A) = \sigma_1\sigma_n^{-1}$ will give the condition number for the 2-norm and can also be used as an estimate of the condition number using other norms. Since a random vector $\underline{b}$ has been used, the probability of a good estimate of $\|A^{-1}\|$ is high. A recent paper [2] describes the strategy of choosing $\underline{b}$ to enhance this probability, and also the estimation of the condition number from the QR and LU form of decomposition.

As a final remark we note that in least squares problems, the matrix A is often m-by-n with $m \gg n$ and hence the normal definition of condition number is not of much use. Instead we use $\sigma_{max}/\sigma_{min}$ as a "condition number" for A, where σ_{max} is the largest singular value and σ_{min} is the smallest non-zero singular value of A. The reason behind this can be seen if we write

$$A\underline{x} = \underline{b} \quad \text{as} \quad A^TA\underline{x} = A^T\underline{b} \ .$$

Digital Image Processing

An image can be represented, in digital form, by a two dimensional array of numbers representing individual brightness values taken from, for example, a camera tube. This matrix representation enables us to "manipulate on the image" in a digit computer with transformations and expansions.

Consider an n-by-n matrix G that represents an image that has been sampled and quantized in space. Let the SVD of G be UDV^T, where D is a diagonal matrix whose diagonal elements are the singular values σ_i of G, that is, $D = \text{diag}(\sigma_1, \ldots, \sigma_n)$. It is clear that if $\sigma_{r+1} = \ldots = \sigma_n = 0$, that is, G is of rank r, then

$$G = \sigma_1 \underline{u}_1 \underline{v}^T + \ldots + \sigma_r \underline{u}_r \underline{v}_r^T .$$

Letting $E_j = \underline{u}_j \underline{v}_j^T$ for $j = 1(1)r$, $G = \sum_{i=1}^{r} \sigma_i E_i$. For each j, E_j is of rank 1 and $\|E_j\|_2 = 1$. Also, since U,V are unitary, $\|E_j\|_F = 1$ for each j. Hence we cannot say some E_j are numerically more important than others.

Let $G_k = \sum_{i=1}^{k} \sigma_i E_i$, where $1 \leqslant k \leqslant r$. Then G_k can be thought of as some matrix approximating G. Moreover, if $\sigma_{k_1} >> \sigma_{k_1+1}$ for some k_1, then we would expect G_{k_1} to be a good approximation to G. In certain cases, this represents a drastic reduction of storage. Instead of storing the n^2 elements of G, we only need $(2k_1 + 1)n$ storage locations. This reduction of storage without a significant loss of information is clearly highly desirable. If the rank is small, we could actually store all the $(2n + 1)r$ elements of the expansion. The SVD technique has also found application in image restoration and it was noted that "the worse the imaging system, the better is the use of SVD techniques, which is intuitively appealing because in poor imaging there is little left of the object for restoration." [1]

The idea of forming approximating matrices via SVD has recently been applied to the study of cryptography [9]. The SVD technique has also been used to solve the general linear system of equations $A\underline{x} = \underline{b}$, where A is an m-by-n matrix, $\underline{x}$ an n-vector and $\underline{b}$ an m-vector. An interesting special case

is when $m = n$ and $\underline{b} = \underline{0}$, in this case we have to find a non-null solution for a system of homogeneous equations.

The updating of the SVD of a matrix after a rank one change is also of great research interest because of the increasing applications of SVD in various areas. An efficient updating method which minimizes the number of operations and storage locations is still to be found.

IV. Conclusion

The rigorous theoretical background of the SVD and its algorithm, coupled with its stability, has proved that the SVD is a powerful technique in the study of theoretical and computational numerical problems. Although computationally SVD may not be as efficient as other factorization methods, it is probably the best tool for solving ill-conditioned problems. The wide range of applicability of SVD also implies more research should be done on various aspects of SVD to increase our understanding of this factorization technique.

Acknowledgement

Special thanks must be given to Mr. R.L. Broughton for introducing me to this subject and for his help in writing this essay. I would also like to thank Dr. I.D. Coope for his comments.

References

[1] Andrews, H.C. and C.L. Patterson, "Outer product expansions and their use in digital image processing." Amer. Math. Monthly **82**, 1, 1975.

[2] Cline, A.K., Moler, C.B., Stewart, G.W. and Wilkinson, J.H., "An Estimate for the Condition Number of a Matrix," SIAM J. Num. Anal. **16**, 368, 1979.

[3] Eckart, C. and Young, G., "A principal axis transformation for non-Hermitian matrices," Bull. Amer. Math. Soc. **45**, 118, 1939.

[4] Forsythe, G.E., Malcolm, M.A. and Moler, C.B., "Computer methods for Mathematical Computations." Englewood Cliffs, N.J., Prentice-Hall, 1977.

[5] Golub, G.H. and Kahan, W., "Calculating the singular values and pseudo-inverse of a matrix." SIAM J. Num. Anal. **2**, 202, 1965.

[6] Golub, G.H. and Reinsch, C., "Singular value decomposition and least squares solutions." Num. Math. **14**, 403,1970.

[7] Jacobs, D.A.H. ed. "The State of the Art in Numerical Analysis." Academic Press, 1977.

[8] Lawson, C.L. and Hanson, R.J., "Solving least squares problems." Englewood Cliffs, N.J., Prentice-Hall, 1974.

[9] Moler, C.B. and Morrison, D.R., "Singular value decomposition and cryptography," 1977, Cited in [4].

[10] Stewart, G.W., "Introduction to Matrix Computations," Academic Press, 1973.

[11] Sylvester, J.J., Messenger of Math **19**, 42, 1889, cited in [3].

[12] Wilkinson, J.H., "Global convergence of tridiagonal QR algorithm with origin shifts." Lin. Alg. and Appl. **1**, 409, 1968.